

## **CLAIMS**

### **1-26. (Canceled)**

**27. (Currently Amended)** A computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) to:

invoke a deployment manifest to obtain manifest metadata about an application for the purpose of installing the application on a client computing system;

receive the manifest metadata about the application;

issue a query of an install state of the client computing system to determine whether a platform necessary to the application is present on the client computing system;

receive the install state of the necessary platform present on the client computing system;

determine whether the application is authorized for installation on the client computing system; and

enable the application to be installed on the client computing system, wherein during the enabled installation, the application is available for use while being installed, wherein installation on the client computing ~~device~~ system comprises:

receiving a request from the client computing system for a resource;

determining if the requested resource is stored locally on the client computing system;

if the requested resource is not stored locally on the client computing system device, determining if the requested resource is an on demand resource or an online resource;

if the requested resource is an on demand resource, caching the requested resource in an application store; [[and]]

if the resource is an online demand resource, caching the requested resource in a transient cache and copying the requested resource into the application store; and

installing the application when a number of resources stored in the application store reaches a threshold number.

**28. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented (API) of claim 27 wherein the manifest metadata includes information sufficient to describe the application.

**29. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the computer-implemented API receives a parameter that identifies the application.

**30. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27

wherein the computer-implemented API invokes a deployed application identity to obtain the manifest metadata about the application.

**31. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the computer-implemented API invokes both a deployment manifest and a deployed application identity to obtain the manifest metadata about the application.

**32. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the computer-implemented API will abort the installation of the application if the platform is not present.

**33. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 32 wherein the computer-implemented API will return error information in conjunction with aborting the installation of the application.

**34. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the computer-implemented API will generate a set of authorization parameters for an authorized application.

**35. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 34 wherein the set of authorization parameters comprise at least permission grants, privacy policy guarantees, and license keys.

**36. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the platform comprises one or more software modules upon which the application depends that are not part of the application.

**37. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 36 wherein the platform further comprises one or more software modules that cannot be installed as part of the installation of the application.

**38. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the platform is identified in an application manifest associated with the application.

**39. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of

claim 27 wherein the computer-implemented API includes verifying a version associated with the platform.

**40. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the determination of the authorization comprises determining whether the installation of the application exceeds a trust level associated with a source of the application.

**41. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the determination of the authorization comprises determining whether the installation of the application violates a privacy policy associated with the client computing system.

**42. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 27 wherein the determination of the authorization comprises determining whether the installation of the application violates a license associated with the application.

**43. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface

(API) of claim 27 wherein the computer-implemented API includes determining if a version of the application already exists on the client computing system.

**44. (Previously Presented)** The computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) of claim 43 wherein the computer-implemented API includes downloading at least one resource associated with the application if the application does not exist on the client computing system.

**45. (Currently Amended)** A computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) to:

invoke a deployment manifest to obtain manifest metadata about an application for the purpose of installing the application on a client computing system;

receive the manifest metadata about the application;

issue a query of an install state of the client computing system to determine whether a platform necessary to the application is present on the client computing system, wherein the installation of the application is aborted if the platform is not present and error information is returned in conjunction with aborting the installation of the application;

receive the install state of the necessary platform present on the client computing system;

determine whether the application is authorized for installation on the client computing system; and

enable the application to be installed on the client computing system, wherein during the enabled installation, the application is available for use while being installed, wherein installation on the client computing ~~device~~ system comprises:

receiving a request from the client computing system for a resource;

determining if the requested resource is stored locally on the client computing system;

if the requested resource is not stored locally on the client computing system ~~device~~, determining if the requested resource is an on demand resource or an online resource;

if the requested resource is an on demand resource, caching the requested resource in an application store; [[and]]

if the resource is an online ~~demand~~ resource, caching the requested resource in a transient cache and copying the requested resource into the application store; and

installing the application when a number of resources stored in the application store reaches a threshold number.

**46. (Currently Amended)** A computer readable storage medium encoded with a computer-implemented Application Programming Interface (API) to:

invoke a deployment manifest to obtain manifest metadata about an application for the purpose of installing the application on a client computing system;

receive the manifest metadata about the application;

issue a query of an install state of the client computing system to determine whether a platform necessary to the application is present on the client computing system;

receive the install state of the necessary platform present on the client computing system;

determine whether the application is authorized for installation on the client computing system, wherein the computer-implemented API will generate a set of authorization parameters for an authorized application comprising at least permission grants, privacy policy guarantees, and license keys; and

enable the application to be installed on the client computing system, wherein during the enabled installation, the application is available for use while being installed, wherein installation on the client computing ~~device~~ system comprises:

receiving a request from the client computing system for a resource;

determining if the requested resource is stored locally on the client computing system;

if the requested resource is not stored locally on the client computing system ~~device~~, determining if the requested resource is an on demand resource or an online resource;



if the requested resource is an on demand resource, caching the requested resource in an application store; [[and]]

if the resource is an online ~~demand~~ resource, caching the requested resource in a transient cache and copying the requested resource into the application store; and

installing the application when a number of resources stored in the application store reaches a threshold number.